CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence is being transmitted by either submission using the EFS WEB submission system, fax to the U.S. Patent and Trademark office to fax number 571-273-8300, or is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 to on November 7, 2007.

/Brian C. Kunzler/
Attorney for Applicant

PATENT
Docket No. TUC920030083US1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Scott Milton Fry et al.	
Serial No.:	10/644,378	
Filed:	August 20, 2003	Group Art Unit: 2129
For:	APPARATUS, SYSTEM, AND METHOD FOR DEVELOPING FAILURE PREDICTION SOFTWARE	
Examiner:	Peter D. Coughlan	

APPEAL BRIEF

Mail Stop Appeal Brief Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Examiner:

The USPTO received Appellants' timely Notice of Appeal on September 7, 2007 which was filed in response to the Final Office Action mailed June 7, 2007. Appellants appeal the rejection of pending Claims 1-7, 9-13, 15-25, and 27-40.

This Appeal Brief is being filed under the provisions of 37 C.F.R. § 41.37. The filing fee set forth in 37 C.F.R. § 41.20(b)(2) of $510.00 is submitted herewith. The Commissioner is hereby authorized to charge payment of any additional fees associated with this communication, or to credit any overpayment, to Deposit Account No. 090449.

1. REAL PARTY IN INTEREST

The real party in interest is the assignee, International Business Machines Corporation, Armonk, New York.

2. RELATED APPEALS AND INTERFERENCES

There are no related appeals, interferences, or judicial proceedings.

3. STATUS OF CLAIMS

The Final Office Action rejected Claims 1-7, 9-13, 15-25, and 27-40. Claims 1-7, 9-13, 15-25, and 27-40 stand rejected under 35 U.S.C. § 101 for nonstatutory subject matter. Claims 1, 3, 13, 18, 21, 25, 30, 32, and 36 stand rejected under 35 U.S.C. § 102(a) as being unpatentable over *Awadallah*, "Application of AI Tools in Fault Diagnosis of Electrical Machines and Drives – An Overview," (hereinafter "*Awadallah*"). Claims 2, 6, 19, 20, 24, 31, 35, 37, and 38 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of U.S. Patent 6,446,081 to *Preston* (hereinafter "*Preston*"). Claims 5, 23, 29, and 34 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of U.S. Patent 4,907,230 to *Heller et al.* (hereinafter "*Heller*"). Claim 39 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of *Preston* and further in view of *Heller*. Claims 4, 22, and 33 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of U.S. Patent 6,314,377 to *Ottesen et al.* (hereinafter "*Ottesen*"). Claim 7 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of U.S. Patent 6,219,805 to *Jones et al.* (hereinafter "*Jones*"). Claims 27, 28 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of U.S. Patent 6,553,369 to *Guay et al.* (hereinafter "*Guay*"). Claim 40 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of *Preston* and *Heller*, and further in view of U.S. Patent 6,397,202 to *Higgens et al.* (hereinafter "*Higgens*").

On August 7, 2007, after the Final Office Action, Appellants submitted an Amendment and Response to Office Action, in which Appellants amended Claims 9 and 15 to comply with a requirement of form expressly set forth in the previous Office Action, depending on a canceled claim, and further to present the claims in better form for consideration on appeal. (*See* Final

Office Action, page 2, lines 6-8). According to MPEP § 1116(b) such amendments should be allowed after a final rejection. The Advisory Action dated August 17, 2007, however, was silent on the amendments, neither entering them nor rejecting them. For purposes of appeal, Appellants assume that the amendments were valid and entered. All claims remain rejected. Appellants appeal the rejection of Claims 1-7, 9-13, 15-25, and 27-40.

4. STATUS OF AMENDMENTS

There are no proposed amendments at issue from the Claims and Specification considered by the Examiner in the Final Office Action. As noted above, On August 7, 2007, after the Final Office Action, Appellants submitted an Amendment and Response to Office Action, in which Appellants amended Claims 9 and 15 to comply with a requirement of form expressly set forth in the previous Office Action. The Advisory Action of August 17, 2007 did not state whether or not the amendments have been entered. Based on the guidelines set forth in MPEP § 116(b), Appellants will assume that the minor amendments have been entered for purposes of appeal. A copy of the Claims, as they stand for purposes of appeal, is presented in the Claims Appendix.

5. SUMMARY OF CLAIMED SUBJECT MATTER

The claimed subject matter deals with developing failure prediction software for a storage system, and deals further with forecasting failure of one or more components of a storage system. (*See*, Abstract, lines 1-2 and 17-20).

The problem addressed by the present invention is the difficulty of developing software to predict the occurrence of permanent storage system errors. (Page 3, lines 13-16). More specifically, many that are familiar with data storage systems and their characteristics do not have the advanced computer programming knowledge necessary to translate their knowledge of storage system failure to computer program code. (Page 3, lines 16-18). Storage system failure prediction algorithms may involve large amounts of performance data, and failure prediction algorithms using discrete threshold values may not accurately predict failure, and may not be customizable to an end user. (Page 3, line 19 – page 4, line 5). Relying on software engineers who may not possess extensive or detailed knowledge of the operation of storage systems to update failure prediction software may also result in lengthy, repeated development cycles. (Page 4, lines 6-13).

Embodiments of the present invention include an apparatus for developing failure prediction software for a storage system, an apparatus for predicting component failure within a storage system, a system for predicting component failure within a storage system, a method for developing failure prediction software for a storage system, a method for predicting component failure within a storage system, another apparatus for developing failure prediction software for a storage system, and an article of manufacture comprising a program storage medium readable by a processor and embodying one or more instructions executable by a processor to perform a method for developing failure prediction software for a storage system.[1] *See*, *e.g.*, Claims 1, 7, 13, 18, 25, 30, and 36.

Claim 1 presents an apparatus for developing failure prediction software for a storage system (page 13, lines 8-11; Fig. 2). The apparatus comprises an editor (Fig. 2, num. 202; page 13, lines 12-20; page 14, lines 1-24), a code generator (Fig. 2, num. 212; page 14, line 25 – page 16, line 7), a test module (Fig. 2, num. 220; page 16, line 8 – page 17, line 3), and a revision module (Fig. 2, num. 224; page 17, lines 4-10).

The following quotation of Claim 1 includes reference numerals and parenthetical references to representative examples of the elements and components recited in Claim 1 in compliance with 37 C.F.R. 41.37 (c)(1)(v).

> 1. An apparatus for developing failure prediction software for a storage system (Fig. 2, num. 200; page 13, lines 8-11), comprising:
>
> an editor to assist a user in generating a failure prediction algorithm (Fig. 2, num. 202; page 13, lines 12-20) comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format (Fig. 2, num. 210; Fig. 6B, num. 616; page 14, lines 1-24; page 27, line 1 – page 32, line 24);
>
> a code generator to generate machine-readable code from the stored failure prediction algorithm in response to user input (Fig. 2, num. 212; page 14, line 25 – page 16, line 7);

[1] Although Appellant has summarized embodiments of the present invention, the present invention is defined by the claims themselves. Appellant's summary is not intended to limit the scope of the claims or individual claim elements in complying with the appeal brief requirements under 37 C.F.R. § 41.37(c)(v).

a test module to test the machine-readable code with sample data to produce a result in response to user input (Fig. 2, num. 220; page 16, line 8 – page 17, line 3); and

a revision module to allow revisions of the failure prediction algorithm in response to user input such that the result corresponds to an expected result (Fig. 2, num. 224; page 17, lines 4-10).

Claim 7 presents an apparatus for predicting component failure within a storage system (page 17, line 26-27; Fig. 3). The apparatus comprises a performance monitor (Fig. 3, num. 302; page 18, lines 1-24), a processor (Fig. 3, num. 304; page 18, line 25 – page 19, line 7), a determination module (Fig. 3, num. 306; page 18, line 8 – page 20, line 16), and an interface (Fig. 3, num. 311; page 20, line 8 – page 21, line 4).

The following quotation of Claim 7 includes reference numerals and parenthetical references to representative examples of the elements and components recited in Claim 7 in compliance with 37 C.F.R. 41.37 (c)(1)(v).

7. An apparatus for predicting component failure within a storage system (Fig. 3, num. 300; page 17, line 26-27), the apparatus comprising:

a performance monitor to gather performance data for a storage system (Fig. 3, num. 302; page 18, lines 1-24);

a processor to execute a failure prediction algorithm on the performance data to produce a result (Fig. 3, num. 304; page 18, line 25 – page 19, line 7), the failure prediction algorithm comprising fuzzy logic rules (Fig. 2, num. 210; Fig. 6B, num. 616; page 14, lines 1-24; page 27, line 1 – page 32, line 24);

a determination module to selectively forecast failure of one or more components of the storage system in response to the result (Fig. 3, num. 306; page 18, line 8 – page 20, line 16); and

an interface to adjust a predefined quality threshold of the determination module in response to user input, thereby adjusting the degree of data loss risk and remedial costs associated with a forecasted failure of one or more components (Fig. 3, num. 311; page 20, line 8 – page 21, line 4).

Claim 13 presents a system for predicting component failure within a storage system (page 32, lines 25-26; Fig. 7). The system comprises a controller (Fig. 7, num. 702; page 33, lines 3-7), a communication module (Fig. 7, num. 704; page 33, lines 7-12), a drive mechanism

(Fig. 7, num. 706; page 33, lines 11-15), and an analysis module (Fig. 7, num. 708; page 33, line 16 – page 34, line 13).

The following quotation of Claim 13 includes reference numerals and parenthetical references to representative examples of the elements and components recited in Claim 13 in compliance with 37 C.F.R. 41.37 (c)(1)(v).

13. A system for predicting component failure within a storage system (page 32, lines 25-26; Fig. 7), the system comprising:

a controller to control and manage data transactions with a host (Fig. 7, num. 702; page 33, lines 3-7);

a communication module to exchange data between the host and a storage media (Fig. 7, num. 704; page 33, lines 7-12);

a drive mechanism to read data from the storage media and write data to the storage media (Fig. 7, num. 706; page 33, lines 11-15); and

an analysis module to execute machine-readable code programmed to selectively predict failure of the storage media and the drive mechanism in response to a result from a failure prediction algorithm comprising fuzzy logic rules and performance data associated with the storage media and the drive mechanism, the machine-readable code comprising an interface to selectively adjust a fuzzy variable definition to tune the failure prediction algorithm in response to user input (Fig. 7, num. 708; page 33, line 16 – page 34, line 13).

Claim 18 presents a method for developing failure prediction software for a storage system (page 22, lines 3-4; Fig. 4). The method comprises generating a failure prediction algorithm (Fig. 4, num. 404; page 22, lines 4-18), generating machine-readable code (Fig. 4, num. 406; page 22, lines 19-27), testing the machine readable code (Fig. 4, num. 408; page 23, lines 1-15), and selectively revising the failure prediction algorithm (Fig. 4, num. 412; page 23, lines 16-23).

The following quotation of Claim 18 includes reference numerals and parenthetical references to representative examples of the elements and components recited in Claim 18 in compliance with 37 C.F.R. 41.37 (c)(1)(v).

18. A method for developing failure prediction software for a storage system (page 22, lines 3-4; Fig. 4), the method comprising:

generating a failure prediction algorithm (Fig. 4, num. 404; page 22, lines 4-18) comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format (Fig. 2, num. 210; Fig. 6B, num. 616; page 14, lines 1-24; page 27, line 1 – page 32, line 24);

generating machine-readable code from the stored failure prediction algorithm (Fig. 4, num. 406; page 22, lines 19-27);

testing the machine-readable code to produce a result (Fig. 4, num. 408; page 23, lines 1-15); and

selectively revising the failure prediction algorithm such that the result corresponds to an expected result (Fig. 4, num. 412; page 23, lines 16-23).

Claim 25 presents a method for predicting component failure within a storage system (page 24, lines 9-10; Fig. 5). The method comprises gathering performance data (Fig. 5, num. 504; page 24, lines 10-18), executing a failure prediction algorithm (Fig. 5, num. 508; page 24, line 24 – page 25, line 5), tuning the failure prediction algorithm (Fig. 5, num. 514, 516; page 25, lines 10-23), and selectively forecasting failure of one or more components (Fig. 5, num. 512; page 25, lines 6-9).

The following quotation of Claim 25 includes reference numerals and parenthetical references to representative examples of the elements and components recited in Claim 25 in compliance with 37 C.F.R. 41.37 (c)(1)(v).

25. A method for predicting component failure within a storage system (page 24, lines 9-10; Fig. 5), the method comprising:

gathering performance data for a storage system (Fig. 5, num. 504; page 24, lines 10-18);

executing a failure prediction algorithm on the performance data to produce a result, the failure prediction algorithm comprising fuzzy logic rules (Fig. 5, num. 508; page 24, line 24 – page 25, line 5);

tuning the failure prediction algorithm by adjusting a fuzzy variable definition (Fig. 5, num. 514, 516; page 25, lines 10-23); and

selectively forecasting failure of one or more components of the storage system in response to the result (Fig. 5, num. 512; page 25, lines 6-9).

Claim 30 presents an apparatus for developing failure prediction software for a storage system (page 13, lines 8-11; Fig. 2). The apparatus comprises means for generating a failure prediction algorithm (Fig. 2, num. 202; page 13, lines 12-20; page 14, lines 1-24). One example of a means for generating a failure prediction algorithm is an editor 202, as shown in Fig. 2. The apparatus further comprises means for generating machine-readable code (Fig. 2, num. 212; page 14, line 25 – page 16, line 7). One example of a means for generating machine-readable code is a code generator 212, as shown in Fig. 2. The apparatus further comprises means for testing the machine-readable code (Fig. 2, num. 220; page 16, line 8 – page 17, line 3). One example of a means for testing the machine-readable code is a test module 220, as is shown in Fig. 2. The apparatus further comprises means for selectively revising the failure prediction algorithm (Fig. 2, num. 224; page 17, lines 4-10). One example of a means for selectively revising the failure prediction algorithm is a revision module 224, as is shown in Fig. 2.

The following quotation of Claim 30 includes reference numerals and parenthetical references to representative examples of the elements and components recited in Claim 30 in compliance with 37 C.F.R. 41.37 (c)(1)(v).

> 30. An apparatus for developing failure prediction software for a storage system (page 13, lines 8-11; Fig. 2), comprising:
>
> means for generating a failure prediction algorithm comprising fuzzy logic rules (Fig. 2, num. 202; page 13, lines 12-20; page 14, lines 1-24), the failure prediction algorithm stored in a natural language format (Fig. 2, num. 210; Fig. 6B, num. 616; page 14, lines 1-24; page 27, line 1 – page 32, line 24);
>
> means for generating machine-readable code from the stored failure prediction algorithm (Fig. 2, num. 212; page 14, line 25 – page 16, line 7);
>
> means for testing the machine-readable code to produce a result (Fig. 2, num. 220; page 16, line 8 – page 17, line 3);
>
> means for selectively revising the failure prediction algorithm such that the result corresponds to an expected result (Fig. 2, num. 224; page 17, lines 4-10).

Claim 36 presents an article of manufacture comprising a program storage medium readable by a processor and embodying one or more instructions executable by a processor to perform a method for developing failure prediction software for a storage system (page 22, lines 3-4; Fig. 4). The method comprises generating a failure prediction algorithm (Fig. 4, num. 404; page 22, lines 4-18), generating machine-readable code (Fig. 4, num. 406; page 22, lines 19-27), testing the machine readable code (Fig. 4, num. 408; page 23, lines 1-15), and selectively revising the failure prediction algorithm (Fig. 4, num. 412; page 23, lines 16-23).

The following quotation of Claim 36 includes reference numerals and parenthetical references to representative examples of the elements and components recited in Claim 36 in compliance with 37 C.F.R. 41.37 (c)(1)(v).

36. An article of manufacture comprising a program storage medium readable by a processor and embodying one or more instructions executable by a processor to perform a method for developing failure prediction software for a storage system (page 22, lines 3-4; Fig. 4), the method comprising:

generating a failure prediction algorithm (Fig. 4, num. 404; page 22, lines 4-18) comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format (Fig. 2, num. 210; Fig. 6B, num. 616; page 14, lines 1-24; page 27, line 1 – page 32, line 24);

generating machine-readable code from the stored failure prediction algorithm (Fig. 4, num. 406; page 22, lines 19-27);

testing the machine-readable code to produce a result (Fig. 4, num. 408; page 23, lines 1-15); and

selectively revising the failure prediction algorithm such that the result corresponds to an expected result (Fig. 4, num. 412; page 23, lines 16-23).

6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

I. Whether the Examiner failed to establish a *prima facie* case of anticipation under 35 U.S.C. § 102(b) for Claims 1, 3, 13, 18, 21, 30, 32, and 36 where the limitations of the claims are not taught or suggested in *Awadallah.*

II. Whether the Examiner failed to establish a *prima facie* case of obviousness under 35 U.S.C. § 103(a) for Claims 2, 4-7, 19, 20, 22-24, 27-29, 31, 33-35, and 37-40 where *Awadallah* is non-analogous art, and further where the limitations of the claims are not taught or suggested within the combinations of *Awadallah, Preston, Heller, Ottesen, Jones, Guay,* and/or *Higgens*;

III. Whether the Examiner failed to establish a *prima facie* case of unpatentability under 35 U.S.C. § 101 for Claims 1-7, 9-13, 15-25, and 27-40 where there is a tangible result; and

IV. Whether natural language may be used as inputs, outputs, or variables in the Specification.

7. ARGUMENT

The Advisory Action mailed August 17, 2007 maintains the final rejection set forth in the Final Office Action mailed June 7, 2007. Examiner's rejections of the Claims are based upon § 102(b), § 103(a), and § 101. The Examiner also objects to the specification. Appellants first address the rejections under § 102(b), then address the rejections under § 103(a), following with the rejections based on § 101, and then with the objection to the specification.

I. <u>The Examiner failed to establish a *prima facie* case of anticipation under 35 U.S.C. § 102(b) because *Awadallah* does not teach every element of Claims 1, 3, 13, 18, 21, 30, 32.</u>

The Examiner, in the Final Office Action, rejected Independent Claims 1, 13, 30, and 36 using similar reasoning regarding some claim elements. Appellants do not believe or acquiesce to any implication that the Independent Claims or their elements are identical in scope. However, for purposes of clarity, since the Examiner has raised some of the same arguments for Independent Claims 1, 13, 30, and 36, Appellants focus the argument on these points. The rejection of Independent Claim 1 is representative of the rejections of Independent Claims 13, 30, and 36, on these points, thus Appellants will focus on Independent Claim 1 as a representative claim. The rejection of Independent Claims 1, 13, 30, and 36 under 35 U.S.C. § 102(b) as being anticipated by *Awadallah* was improper because *Awadallah* does not teach each limitation recited in the claims, either alone or in combination.

"Anticipation under 35 U.S.C. § 102 requires the disclosure in a single piece of prior art of each and every limitation of a claimed invention. ...Whether such art is anticipating is a question of fact." *Apple Computer, Inc. v. Articulate Systems, Inc.* 234 F.3d 14, 20, 57 USPQ2d 1057, 1061 (Fed. Cir. 2000). Appellants submit that *Awadallah* does not disclose each and every limitation of Claims 1, 3, 13, 18, 21, 25, 30, 32, and 36 as required for a rejection under § 102. Appellants also respectfully suggest that perhaps the Final Office Action has mistaken the electrical machine drives of *Awadallah* for data storage drives.

A. *__Awadallah__* __Fails to Teach All of the Claim Limitations__

While non-analogous art is improper for use in a rejection under 35 U.S.C. § 103, non-analogous art may be used in a rejection under 35 U.S.C. § 102, if the non-analogous reference discloses each and every limitation of the claimed invention arranged as in the claims. The non-analogous nature of *Awadallah* and the present invention is discussed below with regards to the rejections under 35 U.S.C. § 103. In general, *Awadallah* teaches fault diagnosis of electrical machines and electrical drives. Drives, in *Awadallah*, are mechanisms by which force or power are transferred in a machine, and are not storage devices. *Awadallah* was published in the *IEEE Transactions on Energy Conversion*, a publication of the IEEE Power Engineering Society, whose focus is on the generation, transmission, distribution, conversion, measurement, and control of electric energy. (*Awadallah*, headings). *Awadallah* "focuses on stator, rotor, eccentricity, and bearing damage faults in induction motors and drive systems." (*Awadallah*, page 245, column 2, lines 22-25). The drive systems of *Awadallah* are "inverter-fed induction motor drives" (*Awadallah*, page 246, column 4, lines 9-10) or other "electronic drive systems" (*Awadallah*, page 248, column 7, line 22). These drive systems transfer power from the induction motors. The authors, Mohamed A. Awadallah and Medhat M. Morcos specialize in "electrical power and machines engineering," "power electronics," "power quality," "high-voltage engineering," and the like. (*Awadallah*, page 251, column 14, lines 14-38). References 1-74 disclosed in *Awadallah* deal with induction motors, other electrical motors, and their drive systems. Like *Awadallah*, none of the 74 references deals with storage systems. Nowhere does *Awadallah* teach failure prediction of storage systems, or even teach storage systems.

Independent Claim 1 is "an apparatus for developing failure prediction software for a **storage system**." (Claim 1, emphasis added). Dependent Claim 3 depends from Claim 1, and discloses a similar apparatus. Independent Claim 13 discloses "a system for predicting component failure within a **storage system**," as well as disclosing "a **storage media**," and "a drive mechanism to **read data from the storage media** and **write data to the storage media**." (Claim 13, emphasis added). Independent Claim 18 discloses "a method for developing failure prediction software for a **storage system**." (Claim 18, emphasis added). Dependent Claim 21 depends from Claim 18, and discloses a similar method. Claims 25, 30, 32, and 36 include similar limitations, each including a **storage system**. Appellants respectfully submit that

Awadallah does not teach a storage system, a storage media, or failure prediction of a storage system.

Appellants also respectfully disagree that *Awadallah* teaches the other limitations of the Claims at issue. For example, Independent Claim 1, as a representative claim states:

> 1. An apparatus for developing failure prediction software for a storage system, comprising:
>
> an editor to assist a user in generating a failure prediction algorithm comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format;
>
> a code generator to generate machine-readable code from the stored failure prediction algorithm in response to user input;
>
> a test module to test the machine-readable code with sample data to produce a result in response to user input; and
>
> a revision module to allow revisions of the failure prediction algorithm in response to user input such that the result corresponds to an expected result.

The Final Office Action suggests that the abstract of *Awadallah* teaches the editor of Claim 1. Appellants fail to find any teachings of an editor, a user, generation of a failure prediction algorithm, or a natural language format in the abstract of *Awadallah* as cited in the Final Office Action. Additionally, the abstract of *Awadallah* teaches the "reduction of the human experts involvement in the diagnosis process," teaching away from the involvement of a human user in induction motor failure prediction, instead teaching the substitution of "human experts" with "modern artificial intelligence (AI) tools," as the title of the article suggests (*Awadallah*, abstract, title).

The Final Office Action further suggests that *Awadallah* teaches the code generator of Claim 1, stating that "'generate machine readable code' of Appellants is produced by the 'computer simulations' of *Awadallah*." (Final Office Action, page 6, lines 2-4). Appellants Claim 1 states "a code generator to generate machine-readable code from the stored failure prediction algorithm in response to user input," while the passage in *Awadallah* cited by the Final Office Action states "these [performance characteristics] were obtained either through computer simulations or more applicably through experimental testing of faulty machines." (*Awadallah*, column 2, lines 2-4). Obtaining performance characteristics of induction motors through computer simulations or through testing of faulty motors is clearly not the equivalent of generating machine-readable code from a failure prediction algorithm comprising fuzzy logic rules that is stored in a natural language format. The computer simulations of *Awadallah* simulate faulty induction motors to obtain performance characteristics and study their attributes. The computer simulations of *Awadallah* do not generate machine readable code from natural language algorithms, but generate performance data of faulty motors.

The Final Office Action further suggests that *Awadallah* teaches the test module of Claim 1. The Final Office Action equates "training" of artificial neural networks in *Awadallah* to the "test module to test the machine-readable code with sample data to produce a result in response to user input" of Claim 1. (Final Office Action, page 6, lines 5-6). Appellants respectfully submit that the training of artificial neural networks taught in *Awadallah* does not teach the testing of machine-readable code with sample data. The "training" of *Awadallah* is a brief reference in a bulleted list of "tasks" that artificial neural networks can perform. *Awadallah* does not teach a test module that tests machine-readable code that is generated from a failure prediction algorithm comprising fuzzy logic rules in connection with the artificial neural networks or otherwise.

The Final Office Action also suggests that *Awadallah* teaches "a revision module configured to allow revisions of the failure prediction algorithm in response to user input such that the result corresponds to an expected result" of Claim 1. (Final Office Action, page 6, lines 6-9). In the reference cited by the Final Office Action, *Awadallah* teaches that "adaptive fuzzy systems utilize the learning capabilities of ANNs or the optimization strength of genetic algorithms to adjust the system parameter set in order to enhance the intelligent system's performance based on a priori knowledge." (*Awadallah*, page 249, column 1, lines 29-48). Appellants respectfully submit that *Awadallah* does not teach adjusting a failure prediction algorithm or anything else in response to user input, as detailed in Claim 1, but teaches adjusting a system parameter set in response to artificial neural networks and genetic algorithms, both of which are unrelated to the present invention and are not user input. Appellants further submit that adjusting a system parameter set is not equivalent to revising a natural language failure prediction algorithm comprising fuzzy logic rules. An algorithm is an organized procedure or method with steps, rules, or the like, while a system parameter is a system variable, constant, quantity, or attribute. Appellants submit that a system parameter is not a step or a rule, but merely a value.

Appellants submit that *Awadallah* does not teach a storage system, an editor, a code generator, a test module, or a revision module as arranged in Claim 1, or similar limitations in Claims 3, 13, 18, 21, 25, 30, 32, and 36. Appellants respectfully request that the rejection of Claims 1, 3, 13, 18, 21, 25, 30, 32, and 36 under 35 U.S.C. §102(b) as being anticipated by *Awadallah* be withdrawn.

II. The Examiner failed to establish a *prima facie* case of anticipation under 35 U.S.C. § 103(a) because the cited references, either alone or in combination, do not teach or suggest all of the limitations of Claims 2, 4-7, 19, 20, 22-24, 27-29, 31, 33-35, and 37-40.

"To establish a *prima facie* case of obviousness...the prior art reference (or references combined) must teach or suggest all the claim limitations." MPEP § 2142. In addition, in determining obviousness, the question is what the combined teachings of the references would suggest to one of ordinary skill in the art. In re Keller, 642 F.2d 413, 425, 208 USPQ 871, 881 (CCPA 1981). Thus, even if the prior art teaches each element, a determination must still be made "whether the prior art made obvious the invention as a whole for which patentability is claimed." Hartness International, Inc. v. Simplimatic Engineering Co., 819 F.2d 1100, 2 U.S.P.Q.2d 1826 (Fed. Cir. 1987). As such, "it is insufficient that the prior art disclose[] the components of the patented device, either separately or used in other combinations; there must be some teaching, suggestion, or incentive to make the combination made by the inventor." Northern Telecom, Inc. v. Datapoint Corp., 908 F.2d 931, 934 (Fed. Cir. 1990). In addition, the cited art must be analogous to the field of endeavor in which the inventor labors. With regards to each of the rejections under §103(a) in the present case, the prior art fails to teach or suggest all of the claim limitations and to make obvious the unique advantages of the present application.

A. ***Awadallah*** **is Non-Analogous Art**

Appellants respectfully submit that Awaddalah is non-analogous art and thus is not a valid reference to cite for a §103 rejection. Determining that a cited reference is non-analogous requires a two-step process. *In re Deminski*, 796 F.2d 436, 441-2 (Fed. Cir. 1986); MPEP § 2141.01(a).I. The first step is to determine if the reference is within the inventor's field of endeavor. *Id.* If so, then the reference is analogous. *Id.* If the reference is not within the inventor's field of endeavor, the second step is to determine if the reference is reasonably pertinent to the particular problem with which the inventor was involved. *Id.*

The first question, whether the reference is in the inventor's field of endeavor is narrow in scope. It is not sufficient that the reference and the claimed invention are both in the computer science art as demonstrated by Wang Laboratories, Inc. v. Toshiba Corp., 993 F.2d 858 (Fed. Cir. 1993). The Wang decision is cited in detail at MPEP 2141.01(a) – ANALOGY IN THE ELECTRICAL ARTS:

"Patent claims were directed to single in-line memory modules (SIMMs) for installation on a printed circuit motherboard for use in personal computers. Reference to a SIMM for an industrial controller was not necessarily in the same field of endeavor as the claimed subject matter merely because it related to memories. Reference was found to be in a different field of endeavor because it involved memory circuits in which modules of varying sizes may be added or replaced, whereas the claimed invention involved compact modular memories. Furthermore, since memory modules of the claims at issue were intended for personal computers and used dynamic random-access-memories, whereas reference SIMM was developed for use in large industrial machine controllers and only taught the use of static random- access-memories or read-only-memories, the finding that the reference was nonanalogous was supported by substantial evidence." MPEP 2141.01(a)

Thus, a reference to a memory module was found not to be in the field of endeavor for an invention relating to SIMMs for installation on a printed circuit motherboard. The fact that the claimed invention was for personal computers rather than industrial computers and for random access memory rather than static memory were sufficient distinctions to remove the claimed invention from the same field of endeavor as the cited reference.

With respect to the present invention, the claims recite an apparatus, system, and method for developing failure prediction software for a storage system, and for predicting component failure within a storage system. This field of endeavor is distinct from *Awadallah* which relates to induction and other electric motors, as described above. The mere fact that *Awadallah* references fuzzy logic is not sufficient to establish the same field of endeavor. This is supported by the MPEP's citation to Wang which teaches that two references that both relate to computer memory are not necessarily analogous simply because both references use the term "memory." MPEP 2141.02(a).

The second part of the two-part test for analogous art requires that the cited reference be reasonably pertinent to the particular problem with which the inventor was involved. "A reference is reasonably pertinent if, even though it may be in a different field from that of the inventor's endeavor, it is one which, because of the matter with which it deals, logically would have commended itself to an inventor's attention in considering his problem." In re Clay, 966 F.2d 656, 659 (Fed. Cir. 1992). To answer this question, the purpose of the reference and the claimed invention are compared.

Awadallah relates to the diagnosis of faults in electric motors. *Awadallah* explicitly states what the problem faced and addressed by *Awadallah* is. *Awadallah* states "of all possible fault types, this paper focuses on stator, rotor, eccentricity, and bearing damage faults in induction motors and drive systems." (*Awadallah*, page 245, column 2, lines 22-24). In contrast, the claimed invention of the Application, and in its Specification, "relates to maintenance and storage of data within a storage system." (page 1, lines 3-5).

The problems at issue are completely different. An induction motor eccentricity fault or the like does not commend itself to the mind of an inventor trying to develop failure prediction of the permanent data errors in the storage systems of the present invention. Permanent data errors may be caused, for example, by "a portion of tape having a longitudinal crease" or other electrical, magnetic, or optical damage to data storage media (page 2, lines 13-17). Under Wang, the fact that the two references examined both dealt with computer memory was not sufficient to find that the references were analogous art. Thus, *Awadallah* is not analogous art and is an improper 35 U.S.C. §103(a) reference.

B. *Awadallah* does not Teach each Element of the Claims

Appellants further submit that even if *Awadallah* where analogous art, that as described above with regards to the rejection under 35 U.S.C. § 102(a), *Awadallah* does not teach or suggest a storage system, an editor, a code generator, a test module, or a revision module. Each of the Examiner's §103(a) rejections is based on *Awadallah*, and Independent Claims 1, 7, 13, 18, 25, 30, and 36 each include these limitations or other similar limitations, therefore Claims 2, 4-6, 19-20, 22-24, 27-29, 31, 33-35, and 37-40 which depend from Independent Claims 1, 7, 13, 18, 25, 30, and 36 are allowable under § 103(a). Appellants also respectfully traverse the rejections based on *Preston, Heller, Ottesen, Jones, Guay*, and *Higgens* for the reasons expressed in the Office Action Response dated April 16, 2007 and for other reasons. For the sake of brevity and because each reference is used in combination with *Awadallah,* those arguments are not repeated here.

Appellants respectfully assert that if the prior art of record so clearly demonstrates the obviousness of the claimed invention, a single reference would teach more than just one or two elements of the claimed invention. However, the formation of the combinations used in the rejections is indicative of impermissible hindsight analysis by the Examiner. The sheer number of references used seems to indicate that the claim terms were used in a key word search of the prior art. For certain claims up to four different references are relied upon Once a key word hit was found, there appears to be little analysis performed to determine the applicability of relevance of the reference. *Awadallah*, for example, which is relied upon for all of the § 102 and § 103 rejections, includes the term "drive" that is often associated with data storage systems, but "drive" in *Awadallah* refers to the power transfer system of electric motors, which are completely unrelated to data errors in storage systems. The four references cited regarding the relatively brief Claim 40, broken up in groups of a few words each, is also indicative of hindsight keyword analysis in support of a § 103 obviousness rejection. Appellants respectfully assert that because such analysis is improper the rejections should be overturned.

Given that *Awadallah, Preston, Heller, Ottesen, Jones, Guay,* and *Higgins* fail to teach or suggest all of the elements recited in Independent Claims 1, 7, 13, 18, 25, 30, and 36 of the present Application, Appellants respectfully submit that Independent Claims 1, 7, 13, 18, 25, 30, and 36 are patentable over *Awadallah, Preston, Heller, Ottesen, Jones, Guay,* and *Higgins.* Given that Dependent Claims 2-6 depend from Claim 1, that Dependent Claims 9-12 depend from Claim 7, that Dependent Claims 15-17 depend from Claim 13, that Dependent Claims 19-24 depend from Claim 18, that Dependent Claims 27-29 depend from Claim 25, that Dependent Claims 31-35 depend from Claim 30, and that Dependent Claims 37-40 depend from Claim 36, Appellants respectfully submit that Claims 2-6, 9-12, 15-17, 19-24, 27-29, 31-35 and 37-40 are also patentable over *Awadallah, Preston, Heller, Ottesen, Jones, Guay,* and *Higgins.* Appellants request that the rejection of Claims 1, 3, 13, 18, 21, 25, 30, 32, and 36 under 35 U.S.C. § 102(a) as being anticipated by *Awadallah* be withdrawn, and that the rejection of Claims 2, 4-7, 19, 20, 22-24, 27-29, 31, 33-35, and 37-40 under 35 U.S.C. § 103(a) as being unpatentable over *Awadallah* in view of *Preston, Heller, Ottesen, Jones, Guay,* and *Higgins* also be withdrawn.

III. The Examiner failed to establish a *prima facie* case of unpatentability under 35 U.S.C. § 101 for Claims 1-7, 9-13, 15-25, and 27-40.

Claims 1-7, 9-13, 15-25, and 27-40 stand rejected under 35 U.S.C. §101 for nonstatutory subject matter. The Final Office Action's position is that Independent Claims 1, 7, 13, 18, 25, 30, and 36 each fail to "set forth a practical application of that §101 judicial exception to produce a real-world result." Appellants respectfully disagree.

The U.S. Supreme Court has declared that Congress chose expansive language in 35 U.S.C. § 101 to include "anything under the sun that is made by man." *Diamond v. Chakrabarty*, 447 U.S. 303, 308-09, 206 USPQ 193, 197 (1980); M.P.E.P § 2106.IV.A, 8th ed, rev. 5 (Aug. 2006). This perspective has also been embraced by the Federal Circuit. M.P.E.P. § 2106.IV.A. The Federal courts have held that 35 U.S.C. § 101 does have certain limits. *Id.* The courts have found that the statutory categories of inventions, machine manufacture, composition of matter, and process, have only limited exceptions: abstract ideas, laws of nature, and natural phenomena. *Id.* Claims 1-7, 9-13, 15-25, and 27-40 clearly are not abstract ideas, laws of nature, or natural phenomena and do not fit within one of these judicially recognized exceptions.

The Final Office Action provides no support for a claim that Claims 1-7, 9-13, 15-25, and 27-40 are non-statutory, and thus has not met the U.S. Patent Office's burden of establishing a *prima facie* case. *See in re Oetiker*, 997 F.2d 1443, 1445, 24 USPQ2d 1442, 1444 (Fed. Cir. 1992) ("The examiner bears the initial burden . . . of presenting a *prima facie* case of unpatentability."); M.P.E.P. § 2106.IV.D.

To determine if an invention is statutory, the M.P.E.P. in section 2106 has provided a guide for examiners in determining if the claimed invention falls within the judicial exceptions to § 101. M.P.E.P. § 2106.IV.C. et seq. Claims 1-7, 9-13, 15-25, and 27-40 in the Application are clearly not natural phenomena or are laws of nature. While there is some math involved in the fuzzy logic rules of Claims 1-7, 9-13, 15-25, and 27-40, the claims do not recite any mathematical equations and do not fall under the abstract idea exception, so a rejection based on §101 is improper.

Even if the present invention was found to include a judicial exception, the inquiry does not end with that finding. M.P.E.P. § 2106.IV.C.1. USPTO personnel must then ascertain the scope of the claim to determine whether it covers either a judicial exception to § 101 or a practical application of a judicial exception. *Id.* A practical application of a judicial exception may well be deserving of patent protection. *Id.*; *see Diamond v. Diehr*, 450 U.S. 175, 209 U.S.P.Q. 1 (1981) at 187. The test laid out for determining if the claimed invention is a practical application of an abstract idea is laid out in section 2106 of the MPEP as a two prong test. M.P.E.P. § 2106.IV.C.2. "A claimed invention is directed to a practical application of a 35 U.S.C. 101 judicial exception when it: (A) "transforms" an article or physical object to a different state or thing; or (B) otherwise produces a useful, concrete and tangible result, ..." *Id.*

While the Appellants maintain that Claims 1-7, 9-13, 15-25, and 27-40 clearly do not fall under any of the judicial exceptions and do not include an abstract idea, even if the Claims did include an abstract idea in the form of a mathematical exception, the Appellants assert that Claims 1-7, 9-13, 15-25, and 27-40 are a practical application and are statutory. Under the second prong of the practical application test, the Claims must produce a useful, concrete, and tangible result. *Id.* The standard for a useful, a tangible, and a concrete result are set forth in § 2016.IV.C.2.(2).

For an invention to be useful, the invention must meet the utility requirement of 35 U.S.C. § 101 and must be (i) specific, (ii) substantial, and (iii) credible. *Id.* The Final Office Action has not provided *prima facie* case or any evidence at all that Claims 1-7, 9-13, 15-25, and 27-40 do not meet the utility requirement of § 101. *Id.* As set forth in Claims 1-7, 9-13, 15-25, and 27-40, the claimed invention creates machine-readable code from a natural language failure prediction algorithm, generates and tests the machine-readable code to produce a result, revises the failure prediction algorithm, and in certain claims, predicts system component failure. This results in a specific, substantial, and credible failure prediction algorithm or a failure prediction, in increased component failure forecasting precision, and in shortened development cycles for failure prediction algorithms that can be made by non-programmers. The Appellants respectfully assert that the Final Office Action has not met its burden of showing that the Claims do not meet the utility requirement of 35 U.S.C. § 101 and that the claimed invention is specific, substantial, and credible.

"The tangible requirement does not necessarily mean that a claim must either be tied to a particular machine or apparatus or must operate to change articles or materials to a different state or thing." M.P.E.P. § 2106.IV.C.2.(2).b). The Appellants respectfully assert that the Final Office Action has not put forth a *prima facie* case that Claims 1-7, 9-13, 15-25, and 27-40 do not provide a tangible result. The claim need only produce a "real-world result." *Id.* Claims 1-7, 9-13, 15-25, and 27-40 provide a real world result. In the prior art, component failure prediction required complicated software and routines written by experienced software engineers, were often inaccurate, and had long release cycles. The prediction algorithms were not customized to an individual user's storage devices or situations. Discrete thresholds caused many false-positives or missed failures. *See* pars. 11-14 of the specification.

In contrast, the invention of Claims 1-7, 9-13, 15-25, and 27-40 allows non-programmers to create custom failure prediction algorithms that use fuzzy logic instead of discrete thresholds. The prediction algorithms are therefore more accurate as well as being user specific. The invention of Claims 1-7, 9-13, 15-25, and 27-40 also tests the prediction algorithm and allows a user to modify it until an expected result is attained. The invention of Claims 1-7, 9-13, 15-25, and 27-40 forecasts failure of one or more components of a storage system. The Appellants respectfully assert that Claims 1-7, 9-13, 15-25, and 27-40 provide a tangible, real-world result.

The "concrete result" analysis section of § 2106.IV.C.2.(2) states that "this question arises when a result cannot be assured. In other words, the process must have a result that can be substantially repeatable or the process must substantially produce the same result again." M.P.E.P. § 2106.IV.C.2.(2).c). The Appellants respectfully assert that the Final Office Action has not put forth a *prima facie* case that Claims 1-7, 9-13, 15-25, and 27-40 do not provide a concrete result. When properly implemented, the invention of Claims 1-7, 9-13, 15-25, and 27-40 will result in certain claims in a machine-readable code failure prediction algorithm that the invention has generated from natural language fuzzy logic rules, or in other claims in a component failure prediction produced from such an algorithm. The Appellants respectfully assert that the invention of Claims 1-7, 9-13, 15-25, and 27-40 is not of the type where results vary and are not repeatable, and that the results of the invention are therefore concrete.

The Appellants respectfully assert that Claims 1-7, 9-13, 15-25, and 27-40 clearly meet the second prong of the practical application test set forth in MPEP § 2106.IV.C.2 and that Claims 1-7, 9-13, 15-25, and 27-40 are statutory matter under 35 U.S.C. § 101. In addition, the Appellants respectfully assert that Claims 1-7, 9-13, 15-25, and 27-40 meet the first prong of the practical application test, in that the Claims transform an article to a different state. Claims 1-7, 9-13, 15-25, and 27-40 transform a natural language algorithm to machine-readable code, and in certain embodiments, to a component failure prediction. The Appellants respectfully assert that Claims 1-7, 9-13, 15-25, and 27-40 are statutory in that the Claims do not fall within a judicial exception to 35 U.S.C. § 101 and, if found to contain an abstract idea, are a practical application of the abstract idea.

The Office Action dated December 14, 2006 states that "a portion of Claim 25 states 'selectively forecasting failure of one or more components' discloses enough information to state what the practical application of the invention is." (Office Action Dec. 14, 2006, page 3, lines 18-21). Accordingly, Appellants amended Claim 7 and Claim 13 in good faith to include the forecasting element of Claim 25 based on the Office Action's statement and on representations by the Examiner in the telephone interview of April 10, 2007. Appellants agree with the Office Action of December 14th that forecasting failure of storage system components is a practical application, and further assert that generating and testing machine-readable code from a natural language failure prediction algorithm is also a practical application.

IV. Objection to the Specification

The Final Office Action objected to the Specification based on the use of "natural language as inputs, outputs or variables" because natural language "depends on the user's knowledge or use of the language to interpret the results." (Final Office Action, page 3, lines 1-6) Applicants respectfully traverse this objection based on the following remarks.

The Specification details the term natural language as used in Claims 1, 18, 30, and 36 in page 27, line 1 to page 20, line 26. The Final Office Action's objection cites the example found in page 29, lines 7-17 of the Specification. In the example, the number of blocks processed by a data storage system without an error is 1,040 blocks. Based on the graph in Figure 6A, a line representing 1,040 blocks on the X axis intersects the "good" line at 0.0, the "marginal" line at 0.178, and the "bad" line at 0.78. The Office Action poses the question "what is worst, 'not good' or 'not very marginal' if the values to these variables are not known?" (Final Office Action, page 3, lines 4-6). In the example, 'not good' and 'not very marginal' both correspond to the same well defined point on the graph, so neither is worst, both are equal, and both are precisely and exactly known. The Advisory Office Action dated August 17, 2007 takes issue with the overlap between bad and marginal and marginal and good in Fig. 6A (page 2, lines 3-5). In the example, however, the words 'pretty,' 'not very,' and 'very' modify 'good,' 'marginal,' and 'bad,' defining the degree of overlap according to predefined criteria.

The natural language "inputs, outputs or variables" as they are referred to in the Office Action, or fuzzy variables or fuzzy sets as they are referred to in the specification, are carefully and precisely defined and tuned by the user or by an expert in the field using lines or functions. (See page 29, line 18 – page 30, line 5). The lines may be defined using sets of tuples, slopes, equations, or otherwise, but the values and ranges of the fuzzy variables are precisely known. Fuzzy logic is a system that makes consistent determinations regarding precise data based on subjective user criteria. Because the user's criteria or the variable names chosen may be subjective (good, marginal, and bad in the example) does not mean that the values and ranges of the criteria are not known. If the values and ranges of the "inputs, outputs or variables" were not known, fuzzy logic could not be used to make a determination. Page 31, lines 8-18 describes how crisp input values are "fuzzified" into their degrees of membership in each of the fuzzy sets so that fuzzy logic rules can be evaluated based on the input values. Because "a failure prediction algorithm comprising fuzzy logic rules" is "stored in a natural language format," as stated in Claims 1, 18, 30, and 36, does not mean that the fuzzy logic rules are unknown or based on "the users knowledge or use of the language" as the Office Action suggests in par. 5. Conversely, to make a valid determination according to the fuzzy logic rules, each of the "inputs, outputs or variables" should be precisely defined, as detailed in the Specification. Applicants respectfully traverse the objection to the specification, and submit that the specification is allowable based on the above remarks.

SUMMARY

In view of the foregoing, Appellants respectfully assert that each of the claims on appeal has been improperly rejected because the Examiner has not established a *prima facie* case of anticipation under 35 U.S.C. § 102(b) for Claims 1, 3, 13, 18, 21, 30, 32, and 36. Similarly, Appellants assert that the Examiner has not established a *prima facie* case of obviousness under 35 U.S.C. § 103(a) for Claims 2, 4-7, 19, 20, 22-24, 27-29, 31, 33-35. Appellants further assert that the Examiner has not established a *prima facie* case of unpatentability under 35 U.S.C. § 101 for Claims 1-7, 9-13, 15-25, and 27-40. Appellants also assert that natural language may be used as inputs, outputs, or variables in the Specification. Therefore, Appellant respectfully requests reversal of the Examiner's rejections under 35 U.S.C. § 102(a), § 103(a), and § 101, and urges that pending Claims 1-7, 9-13, 15-25, and 27-40 are ready for prompt allowance.

Respectfully submitted,

Date: November 7, 2007

/Brian C. Kunzler/

Brian C. Kunzler
Reg. No. 38,527
Attorney for Applicant

Kunzler & McKenzie
8 E. Broadway, Suite 600
Salt Lake City, Utah 84111
Telephone: 801/994-4646

8. CLAIMS APPENDIX

The claims involved in the appeal, namely Claims 1-40, are listed below.

1. (Previously Presented) An apparatus for developing failure prediction software for a storage system, comprising:

   an editor to assist a user in generating a failure prediction algorithm comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format;

   a code generator to generate machine-readable code from the stored failure prediction algorithm in response to user input;

   a test module to test the machine-readable code with sample data to produce a result in response to user input; and

   a revision module to allow revisions of the failure prediction algorithm in response to user input such that the result corresponds to an expected result.

2. (Original) The apparatus of claim 1, wherein the fuzzy logic rules comprise linguistic variables having less than four terms.

3. (Previously Presented) The apparatus of claim 1, wherein the test module further tunes the failure prediction algorithm by adjusting a fuzzy variable definition in response to user input.

4. (Original) The apparatus of claim 1, wherein the machine-readable code is configured to execute on a storage system.

5. (Previously Presented) The apparatus of claim 1, wherein the revision module comprises a text editor to revise the failure prediction algorithm in response to user input.

6. (Original) The apparatus of claim 1, wherein the fuzzy logic rules are defined by conditional statements that include subjects, adjectives, and verbs familiar to personnel in the storage system field.

7. (Previously Presented) An apparatus for predicting component failure within a storage system, the apparatus comprising:

a performance monitor to gather performance data for a storage system;

a processor to execute a failure prediction algorithm on the performance data to produce a result, the failure prediction algorithm comprising fuzzy logic rules;

a determination module to selectively forecast failure of one or more components of the storage system in response to the result; and

an interface to adjust a predefined quality threshold of the determination module in response to user input, thereby adjusting the degree of data loss risk and remedial costs associated with a forecasted failure of one or more components.

8. (Canceled)

9. (Previously Presented) The apparatus of claim 7, further comprising an interface to adjust a fuzzy variable definition to tune the failure prediction algorithm in response to user input.

10. (Original) The apparatus of claim 9, further comprising a pre-processor to pre-process performance data to provide input data for the failure prediction algorithm.

11. (Previously Presented) The apparatus of claim 10, wherein the determination module maps the result from the failure prediction algorithm to one of a plurality of predefined recommendations.

12. (Previously Presented) The apparatus of claim 11, further comprising a notification module to produce a notification in response to the result.

13. (Previously Presented) A system for predicting component failure within a storage system, the system comprising:

a controller to control and manage data transactions with a host;

a communication module to exchange data between the host and a storage media;

a drive mechanism to read data from the storage media and write data to the storage media; and an analysis module to execute machine-readable code programmed to selectively predict failure of the storage media and the drive mechanism in response to a result from a failure prediction algorithm comprising fuzzy logic rules and performance data associated with the storage media and the drive mechanism, the machine-readable code comprising an interface to selectively adjust a fuzzy variable definition to tune the failure prediction algorithm in response to user input.

14. (Canceled)

15. (Previously Presented) The system of claim 13, wherein the machine-readable code further comprises a pre-processor to pre-process performance data to provide input data for the failure prediction algorithm.

16. (Previously Presented) The system of claim 15, wherein the machine-readable code further comprises a determination module to map a result from the failure prediction algorithm to one of a plurality of predefined recommendations.

17. (Previously Presented) The system of claim 16, wherein the machine-readable code further comprises a notification module to produce a notification in response to the result.

18. (Previously Presented) A method for developing failure prediction software for a storage system, the method comprising:

generating a failure prediction algorithm comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format;

generating machine-readable code from the stored failure prediction algorithm;

testing the machine-readable code to produce a result; and

selectively revising the failure prediction algorithm such that the result corresponds to an expected result.

19. (Original) The method of claim 18, wherein the fuzzy logic rules comprise linguistic variables having less than four terms.

20. (Original) The method of claim 18, wherein certain linguistic variables comprise less than three terms.

21. (Original) The method of claim 18, further comprising tuning the failure prediction algorithm by adjusting a fuzzy variable definition.

22. (Original) The method of claim 18, wherein the machine-readable code is configured to execute on a storage system.

23. (Original) The method of claim 18, further comprising revising the failure prediction algorithm by way of a text editor.

24. (Original) The method of claim 18, wherein the fuzzy logic rules are defined by conditional statements that include subjects, adjectives, and verbs familiar to personnel in the storage system field.

25. (Previously Presented) A method for predicting component failure within a storage system, the method comprising:

gathering performance data for a storage system;

executing a failure prediction algorithm on the performance data to produce a result, the failure prediction algorithm comprising fuzzy logic rules;

tuning the failure prediction algorithm by adjusting a fuzzy variable definition; and

selectively forecasting failure of one or more components of the storage system in response to the result.

26. (Canceled)

27. (Original) The method of claim 25, further comprising mapping the result to one of a plurality of predefined recommendations.

28. (Original) The method of claim 25, further comprising producing a notification in response to the result.

29. (Original) The method of claim 25, further comprising pre-processing performance data to provide input data for the failure prediction algorithm.

30. (Previously Presented) An apparatus for developing failure prediction software for a storage system, comprising:

means for generating a failure prediction algorithm comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format;

means for generating machine-readable code from the stored failure prediction algorithm;

means for testing the machine-readable code to produce a result;

means for selectively revising the failure prediction algorithm such that the result corresponds to an expected result.

31. (Original) The apparatus of claim 30, wherein the fuzzy logic rules comprise linguistic variables having less than four terms.

32. (Original) The apparatus of claim 30, further comprising means for tuning the failure prediction algorithm by adjusting a fuzzy variable definition.

33. (Original) The apparatus of claim 30, wherein the machine-readable code is configured to execute on a storage system.

34. (Original) The apparatus of claim 30, further comprising means for revising the failure prediction algorithm by way of a text editor.

35. (Original) The apparatus of claim 30, wherein the fuzzy logic rules are defined by conditional statements that include subjects, adjectives, and verbs familiar to personnel in the storage system field.

36. (Previously Presented) An article of manufacture comprising a program storage medium readable by a processor and embodying one or more instructions executable by a processor to perform a method for developing failure prediction software for a storage system, the method comprising:

generating a failure prediction algorithm comprising fuzzy logic rules, the failure prediction algorithm stored in a natural language format;

generating machine-readable code from the stored failure prediction algorithm;

testing the machine-readable code to produce a result;

selectively revising the failure prediction algorithm such that the result corresponds to an expected result.

37. (Original) The article of manufacture of claim 36, wherein the fuzzy logic rules comprise simple conditional statements that include subjects, adjectives, and verbs that are commonly used to describe error conditions of a storage system.

38. (Original) The article of manufacture of claim 37, wherein the method further comprises tuning the failure prediction algorithm by adjusting a fuzzy variable definition.

39. (Original) The article of manufacture of claim 38, wherein the method further comprises revising the failure prediction algorithm by way of a text editor.

40. (Original) The article of manufacture of claim 39, wherein revising the failure prediction algorithm comprises adding fuzzy logic rules to the failure prediction algorithm.

9. EVIDENCE APPENDIX

There is no material to be included in the Evidence Appendix.

10. RELATED PROCEEDINGS APPENDIX

There is no material to be included in the Related Proceedings Appendix.